**CS3244 Project Proposal**

**Group 4**

**Group members**:  
Park Seunghwi A0237123N,  
Wang Mingchuan A0240129N ,  
Le Ha Dang A0277185R,  
Lim Zhi Chao A0252895N,  
Kim Seowoo A0237998B,  
Keagan Lai Yong An A0265953R

**Mentor’s name**: Qihao

**Dataset chosen and description**

Write in your own words (a few sentences) what this dataset is about.

1. *Does your description specify features and instances?*
2. *Do you describe the source and reliability of the data?*

The Fashion-MNIST dataset is a collection of 70,000 greyscale images that is partitioned into a test set of size 60,000 and a training set of size 10,000. Each image is labelled with one of the 10 classes, consisting of 10 different clothings.

Each instance is a 28x28 pixel grayscale image. The data comes in a csv file, where each image is represented by a row consisting of a label column and 784 pixel columns. Each pixel takes a value from 0 - 255, where a higher number represents a darker pixel.

This dataset is heavily inspired by the original MNIST dataset often used for training models. In fact, Zalando Research has created this dataset in the hopes of replacing the MNIST dataset. Fashion-MNIST is a large dataset with the same structure as MNIST, but gives us the opportunity to train a reliable model on a more challenging image classification problem.

**Project Title**

**"Hybrid Image Classification: Evaluating Classical ML vs. Deep Learning"**

**Motivation** Explain why this project is interesting and important and the problem you aim to address.

1. *Does your motivation clearly describe a problem?*
2. *Does it justify the problem’s significance? What are the benefits of addressing this problem? Who benefits from solving it?*

Training a reliable image classification model, in the context of clothing, addresses a fundamental task in the domains of e-commerce, inventory management, or even media and entertainment.

Traditional MNIST datasets, while reliable, do not address the complexity and nuance of real-world classification problems where the shape, texture, and style of clothes vary significantly.

Our project aims to develop and evaluate a reliable machine learning model that can accurately classify fashion items from images. The implementation of our model in the abovementioned domains can create better shopping algorithms to improve the customer experience, automated inventory management, faster categorisation for warehouses and media outlets, and trend detection & analysis.

**General Approach** A high-level description of the general approach you’ll use to address the questions.  Sketch out how you plan to run the necessary analysis and experiments.

1. *What steps will you take in tackling your project?*
2. *For instance, how will you perform data analysis/cleaning? What machine learning models will you consider, and why do you think they are suitable for the problem?*
3. Models to consider:
   1. Deep Neural Network
      1. DNN v1 (CNN)
      2. DNN v2 (CNN + Skip connection (residual network))
      3. DNN v3 (CNN + Skip connection + Transformer)
      4. DNN v4 (Autoencoder)
   2. Feature Based ML
      1. KNN
      2. Decision Tree (Random forest)
      3. SVM
      4. Logistic Regression
      5. XGBoost
4. Data analysis and cleaning
   1. General
      1. Data Normalisation
      2. Data structure analysis
      3. Display random sample and class-wise grid visualisation
      4. Remove blur or unrecognizable images
      5. Class imbalance analysis (Check if classes are imbalanced, make sure each set also has a similar proportion of classes)
   2. Data manipulation
      1. Deep Neural Network
         1. Data augmentation (rotate, flip, crop, translate)
      2. Feature Based ML
         1. SIFT (Scale-Invariant Feature Transform)
         2. HOG (Histogram of Oriented Gradient)
         3. Dimension reduction using PCA
5. Model Implementation
   1. Deep Neural Network
      1. Tensorflow/Pytorch implementation of DNN model using builtin functions like ReLU, Maxpool, Convolution layers.
      2. Hyperparameter tuning using keras wandb etc
   2. Sci-kit-learn implementation of Feature Based ML (KNN, SVM, etc..)
      1. Tryout with different kinds of dataset (original, manipulated dataset)

1. Compile, Train and Evaluate model on test set
   1. Use N-fold technique to split the data
2. Plot illustrations: Training and Validation accuracy, Confidence, Loss graphs, Confusion matrix, Precision Recall F1 Score
3. Using illustrations as actionable insights - consider if we need to augment data to reduce overfitting to the given dataset (EG reduce learning rate, L1/2 regularisation, take average of 4 pixel).
4. Repeat 4 ~ 6 until we get

1. Plot illustrations again based on evaluation matrices

**Evaluation** Indicate how you will evaluate your project, i.e., how will you evaluate how your experiments have turned out?

1. *How will you evaluate how well you did as a whole? Of course the most important aspect is evaluating how well the models you proposed perform. How will you do so? What evaluation metrics will you consider?*

Evaluation metrics:

* Model-wise (inference time /training time /model size & no.parameter)
* Accuracy (percentage of correctly predicted vs actual label)
* F1 Score
* Confusion matrix
* Log loss
* Check for overfitting using training accuracy vs test accuracy
* Decent inference time and memory usage
* Class accuracy
* Top-k Accuracy (percentage of cases where the class is among top k predicted class)
* AUC-ROC, AUC-PR

**Resources** A list of resources you need to conduct the project. This includes additional reading, software, compute, additional datasets, reference code (GitHub links etc) beyond your chosen dataset.

1. *If the resources are public, provide references. If they are not public, how will you plan to obtain them? For example, will the team plan to scrape the data off the internet themselves?*

Readings:

1. ResNet paper: <https://arxiv.org/pdf/1512.03385>
2. ResNet implementation with torch: <https://pytorch.org/hub/pytorch_vision_resnet/>
3. Transformer paper: <https://arxiv.org/pdf/1706.03762>
4. Conda Virtual Environment: <https://docs.conda.io/projects/conda/en/stable/user-guide/tasks/manage-environments.html>
5. SOC Computing GPU cluster: <https://dochub.comp.nus.edu.sg/cf/guides/compute-cluster/start>
6. HOG / SVM

<https://www.ijitee.org/wp-content/uploads/papers/v8i5/E3075038519.pdf>

**Role Assignment** and Schedule. Provide a schedule of work for the entire team indicating when you plan to complete components of the project. Make sure the schedule is plausible.

1. *A tentative schedule indicating when the team plans to complete major milestones.*
2. *This must include assignment of roles to different members. For example, Alan is in charge of data cleaning, Sarah is in charge of initial model exploration. It is fine for multiple members to work on the same tasks, as long as there is a fair distribution of work. The purpose of this is to avoid cases of freeloading.*
3. *You must submit at least two updates to your TA throughout the project. Indicate the dates of these updates (ensure they are reasonably spread out) and the milestones you expect to be completed by then in the next section.*
4. *Make sure you have enough time in your team schedule at the end to gather everyone’s work together to gather insights through discussion and by comparing and contrasting the datasets and algorithms.*

**Each student needs to specify two milestones here to ensure that you work in a timely manner.**

**At each of the milestone deadlines, you are required to do a brief update on your progress in the project discussion group on Canvas. The project mentor (TA) will check your progress submissions to ensure that you have contributed to the project as part of evaluation.**

**Please insert your milestones below.**

1. Data Preprocess and cleaning

| Name | Milestone | **Deadline** |
| --- | --- | --- |
| Mingchuan | Perform general exploratory data analysis (EDA) | **23 March 2025** |
| Keagan | Feature extraction\_HOG/SIFT (for basic ML) | **23 March 2025** |
| Dung | Dimension reduction\_PCA (for basic ML) | **23 March 2025** |
| Seunghwi | Data augmentation (for DNN) | **23 March 2025** |

1. Model Implementation

* Complete basic model structure with code regardless of its performance
* Trial and error to find the best hyperparameter or datatype (original or modified version) for each model which gives the best result by repeat training

| Name | Milestone | **Deadline** |
| --- | --- | --- |
| Mingchuan | Baseline model: KNN & evaluation | **30 March 2025** |
| Zhi Chao | Baseline model: Decision Tree & evaluation | **30 March 2025** |
| Keagan, Zhi Chao | Baseline model: SVM & evaluation | **30 March 2025** |
| Dung | Baseline model: Logistic Regression & evaluation | **30 March 2025** |
| Seowoo | Baseline model: XGBoost & evaluation | **30 March 2025** |
| Seunghwi | Baseline model: DNN v1 & evaluation | **30 March 2025** |
| Mingchuan, Seowoo | DNN v2 & evaluation | **30 March 2025** |
| Keagan, Zhi Chao | DNN v3 & evaluation | **30 March 2025** |
| Seunghwi | Autoencoder: DNN v4 & evaluation | **30 March 2025** |

1. Evaluation & Summary

* Gather all the model together and summarize the comparison of the models based on its performances for metrics (Accuracy, F1 … etc)
* Create Video based on the summary

| Name | Milestone | **Deadline** |
| --- | --- | --- |
| Based on assigned | Tune hyperparameters for each models | **06 Apr 2025** |
| All together | Compare all models based on different metrics | **13 Apr 2025** |
| All together | Summarize all the projects in the documents | **13 Apr 2025** |
| All together | Make slides for the presentation and timeline for the video | **13 Apr 2025** |
| All together | Create a final presentation video | **20 Apr 2025 (23:59)** |